ESA instellen om PFS aan te passen

Inhoud

Inleiding

In dit document wordt beschreven hoe u de voorkeur voor Perfect Forward SecRITY (PFS) kunt configureren in gecodeerde verbindingen op Transport Layer Security (TLS) op de E-mail security applicatie (ESA).

Voorwaarden

Vereisten

Cisco raadt u aan kennis te hebben over Secure Socket Layer (SSL)/TLS.

Gebruikte componenten

De informatie in dit document is gebaseerd op AsyncOS voor e-mail versie 9.6 en hoger.

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u de potentiële impact van elke opdracht begrijpen.

Achtergrondinformatie

Het ESR biedt termijngeheimhouding aan. Forwards geheim betekent dat de gegevens worden doorgegeven via een kanaal dat symmetrische encryptie met letterlijke geheimen gebruikt, en zelfs als de private key (lange-termijnsleutel) op een of beide hosts gecompromitteerd is, is het niet mogelijk een eerder opgenomen sessie te decrypteren.

Het geheim wordt niet via het kanaal overgedragen, maar het gedeelde geheim wordt afgeleid van een wiskundig probleem (Diffie Hellman (DH) Probleem). Het geheim wordt nergens anders opgeslagen dan de gastheren Random Access Memory (RAM) tijdens de ingestelde sessie of de belangrijke regeneratietijd.

De ESA ondersteunt DH voor Key Exchange.

Configureren

INBOUND - ESA treedt op als TLS - server

Deze algoritme-series zijn beschikbaar in het ESR voor INBOUND Simple Mail Transfer Protocol (MTP) - verkeer dat doorsturen geheimhouding biedt. In dit voorbeeld laat de selectie van een algoritme alleen algoritme toe die als HOOG of MEDIUM worden beschouwd en gebruikt Ephemeral Diffie Hellman (EDH) voor Key Exchange en geeft de voorkeur aan TLSv1.2. De syntaxis van de algoritmische selectie volgt de syntaxis van OpenSSL.

Knipperaars met voorwaartse geheimhouding op AsyncOS 9.6+:

```
"EDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:!LOW:!EXP:!aNULL:!RC4:!DSS:!SEED:!IDEA:!MD5:!PSK:!3DES:!SRP"

List:  DHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(256) Mac=AEAD
DHE-RSA-AES256-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(256) Mac=SHA256
DHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES128-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(128) Mac=SHA256
DHE-RSA-AES256-SHA SSLv3 Kx=DH Au=RSA Enc=AES(256) Mac=SHA1
DHE-RSA-CAMELLIA256-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(256) Mac=SHA1
DHE-RSA-AES128-SHA SSLv3 Kx=DH Au=RSA Enc=AES(128) Mac=SHA1
DHE-RSA-CAMELLIA128-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(128) Mac=SHA1
```

Het KX (= Key Exchange) - gedeelte toont aan dat DH wordt gebruikt om het geheim af te leiden.

Het ESA ondersteunt deze ciphers met de standaard **slinslinkings** instellingen (:ALL), maar geeft er geen voorkeur aan. Als u liever ciphers wilt hebben die PFS bieden, moet u uw **slineconfig** wijzigen en EDH of een combinatie **EDH+<algoritme of algoritme groepsnaam>** aan uw algoritselectie toevoegen.

Standaardconfiguratie:

```
ESA> sslconfig

sslconfig settings:
 Inbound SMTP method:  tlsv1/tlsv1.2
 Inbound SMTP ciphers:
       RC4-SHA
       RC4-MD5
       ALL
```

Nieuwe configuratie:

```
ESA> sslconfig
```

```
Inbound SMTP method:  tlsv1/tlsv1.2
 Inbound SMTP ciphers:
       EDH+TLSv1.2
       EDH+HIGH
       EDH+MEDIUM
       RC4-SHA
       RC4-MD5
       ALL
```

Opmerking: RC4 als algoritme en MD5 als MAC wordt als zwak beschouwd, als nalatenschap en om het gebruik met SSL/TLS te voorkomen, vooral wanneer het gaat om een hoger gegevensvolume zonder essentiële regeneratie.

Aanbevolen configuratie-instellingen voor INBOUND

Dit is een heersende mening en het is alleen maar mogelijk dat er ciphers komen die algemeen als sterk en veilig worden beschouwd.

Een aanbevolen configuratie voor INBOUND die RC4- en MD5-alsmede andere erfenis- en zwakke opties verwijdert, namelijk Exporteren (EXP), Laag (LOW), IDEA (IDEA), SEED (SEED), 3DES (3DES)-telefoons, DSS-certificaten (DSS), anonieme Key Exchange (NULL), pre-Shared Keys (PSK), SRP-protocol P), schakelt Elliptic Curve Diffie Hellman (ECDH) in voor Key Exchange en Elliptic Curve Digital Signature Algorithm (ECDSA), als volgt uit:

```
EDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:HIGH:MEDIUM:!ECDH:!ECDSA:!LOW:!EXP:!aNULL:!RC4:!DSS:!SEED:!IDEA:
!MD5:!PSK:!3DES:!SRP
```

Het string die in **slfig** is ingevoerd resulteert in deze lijst met ondersteunde tekens voor INBOUND:

```
DHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(256) Mac=AEAD
DHE-RSA-AES256-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(256) Mac=SHA256
DHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES128-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(128) Mac=SHA256
DHE-RSA-AES256-SHA SSLv3 Kx=DH Au=RSA Enc=AES(256) Mac=SHA1
DHE-RSA-CAMELLIA256-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(256) Mac=SHA1
DHE-RSA-AES128-SHA SSLv3 Kx=DH Au=RSA Enc=AES(128) Mac=SHA1
DHE-RSA-CAMELLIA128-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(128) Mac=SHA1
AES256-GCM-SHA384 TLSv1.2 Kx=RSA Au=RSA Enc=AESGCM(256) Mac=AEAD
AES256-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AES(256) Mac=SHA256
AES256-SHA SSLv3 Kx=RSA Au=RSA Enc=AES(256) Mac=SHA1
CAMELLIA256-SHA SSLv3 Kx=RSA Au=RSA Enc=Camellia(256) Mac=SHA1
AES128-GCM-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AESGCM(128) Mac=AEAD
AES128-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AES(128) Mac=SHA256
AES128-SHA SSLv3 Kx=RSA Au=RSA Enc=AES(128) Mac=SHA1
CAMELLIA128-SHA SSLv3 Kx=RSA Au=RSA Enc=Camellia(128) Mac=SHA1
```

Opmerking: Het ESA dat als TLS server (INBOUND-verkeer) fungeert steunt momenteel geen Elliptic Curve Diffie Hellman voor Key Exchange (ECDHE) en ECDSA-certificaten.

OUTBOUND - ESA werkt als TLS-client

Voor buiten het verkeer van MTP, steunt het ESA naast INBOUND ECDHE en ECDSA Certificaten.

> Opmerking: Ecryptografie (ECC) - certificaten met de ECDSA worden niet op grote schaal gebruikt.

Wanneer een OUTBOUND-e-mail wordt afgeleverd, is de ESA de TLS-client. Een TLS-client-certificaat is optioneel. Indien de TLS-server de ESA (als TLS-client) niet dwingt (verplicht) om een ECDSA-clientcertificaat te verstrekken, kan de ESA doorgaan met een ECDSA-beveiligde sessie. Wanneer de ESA als TLS-client om zijn certificaat wordt gevraagd, levert deze het geconfigureerde RSA-certificaat voor de OUTBOUND-richting.

> **Voorzichtig:** De vooraf geïnstalleerde Trusted CA certificaatwinkel (systeemlijst) op de ESA bevat geen ECC (ECDSA) Root Certificates! Het kan nodig zijn om ECC Root Certificates (dat u vertrouwt) handmatig aan de Aangepaste Lijst toe te voegen om de ECC-keten van vertrouwen verifieerbaar te maken.

Om de voorkeur te geven aan DHE/ECDHE-ciphers die Doorsturen geheimhouding aanbieden, kunt u de selectie van het slingeconfig als volgt wijzigen.

Voeg dit toe aan de huidige selectie van het algoritme.

```
"EDH+TLSv1.2:ECDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:ECDH+HIGH:ECDH+MEDIUM"
```

Aanbevolen configuratie-instellingen voor OUTBOUND

Dit is een heersende mening en het is alleen maar mogelijk dat er ciphers komen die algemeen als sterk en veilig worden beschouwd.

```
EDH+TLSv1.2:ECDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:ECDH+HIGH:ECDH+MEDIUM:HIGH:MEDIUM:!LOW:!EXP:!aNULL:
!RC4:!DSS:!SEED:!IDEA:!MD5:!PSK:!3DES:!SRP
```

Het string die in **slfig** is ingevoerd resulteert in deze lijst met ondersteunde ciphers voor OUTBOUND:

```
DHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(256) Mac=AEAD
DHE-RSA-AES256-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(256) Mac=SHA256
DHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES128-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(128) Mac=SHA256
ECDHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH Au=RSA Enc=AESGCM(256) Mac=AEAD
ECDHE-ECDSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AESGCM(256) Mac=AEAD
ECDHE-RSA-AES256-SHA384 TLSv1.2 Kx=ECDH Au=RSA Enc=AES(256) Mac=SHA384
ECDHE-ECDSA-AES256-SHA384 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AES(256) Mac=SHA384
ECDHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=ECDH Au=RSA Enc=AESGCM(128) Mac=AEAD
ECDHE-ECDSA-AES128-GCM-SHA256 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AESGCM(128) Mac=AEAD
ECDHE-RSA-AES128-SHA256 TLSv1.2 Kx=ECDH Au=RSA Enc=AES(128) Mac=SHA256
ECDHE-ECDSA-AES128-SHA256 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AES(128) Mac=SHA256
DHE-RSA-AES256-SHA SSLv3 Kx=DH Au=RSA Enc=AES(256) Mac=SHA1
DHE-RSA-CAMELLIA256-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(256) Mac=SHA1
DHE-RSA-AES128-SHA SSLv3 Kx=DH Au=RSA Enc=AES(128) Mac=SHA1
DHE-RSA-CAMELLIA128-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(128) Mac=SHA1
ECDHE-RSA-AES256-SHA SSLv3 Kx=ECDH Au=RSA Enc=AES(256) Mac=SHA1
ECDHE-ECDSA-AES256-SHA SSLv3 Kx=ECDH Au=ECDSA Enc=AES(256) Mac=SHA1
ECDHE-RSA-AES128-SHA SSLv3 Kx=ECDH Au=RSA Enc=AES(128) Mac=SHA1
ECDHE-ECDSA-AES128-SHA SSLv3 Kx=ECDH Au=ECDSA Enc=AES(128) Mac=SHA1
```

```
AES256-GCM-SHA384 TLSv1.2 Kx=RSA Au=RSA Enc=AESGCM(256) Mac=AEAD
AES256-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AES(256) Mac=SHA256
AES256-SHA SSLv3 Kx=RSA Au=RSA Enc=AES(256) Mac=SHA1
CAMELLIA256-SHA SSLv3 Kx=RSA Au=RSA Enc=Camellia(256) Mac=SHA1
AES128-GCM-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AESGCM(128) Mac=AEAD
AES128-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AES(128) Mac=SHA256
AES128-SHA SSLv3 Kx=RSA Au=RSA Enc=AES(128) Mac=SHA1
CAMELLIA128-SHA SSLv3 Kx=RSA Au=RSA Enc=Camellia(128) Mac=SHA1
```

Verifiëren

Er is momenteel geen verificatieprocedure beschikbaar voor deze configuratie.

Problemen oplossen

Er is momenteel geen specifieke troubleshooting-informatie beschikbaar voor deze configuratie.

Gerelateerde informatie

- Open SSL-ciphers
- Cisco-encryptie van de volgende generatie
- Technische ondersteuning en documentatie – Cisco Systems